

CLAIMS

What is claimed is:

1. A data processing system for processing programs, the system comprising:
memory for storing data objects, the data objects being referenced by
pointers; and
a short-quasi-unique-identifier (SQUID) generator which generates
SQUIDs for newly allocated data objects to be stored in the memory segment,
pointers to a particular data object being associated with the data object's
SQUID.
2. The system of Claim 1, further comprising:
a memory allocator which allocates a segment of the memory to a data
object.
3. The system of Claim 2, wherein if the data object is moved to a second allocated
memory segment, a pointer to the second allocated memory segment is placed at
the original memory segment.
4. The system of Claim 3, wherein the data object is moved due to resizing.
5. The system of Claim 3, wherein the data object is moved from a first memory to
a second memory within a distributed system.
6. The system of Claim 3, wherein the data object is moved due to garbage
collection.

7. The system of Claim 3, wherein the data object is moved due to data compaction.
8. The system of Claim 1, wherein the distribution of SQUIDs over a range is uniform.
- 5 9. The system of Claim 8, wherein SQUIDs are generated by counting.
10. The system of Claim 8, wherein SQUIDs are generated randomly.
11. The system of Claim 8, wherein SQUIDs are generated by hashing.
12. The system of Claim 1, further comprising:
a comparator which compares SQUIDs associated with two different
10 pointers.
13. The system of Claim 12, further comprising:
an instruction reordering mechanism which reorders instructions and
which is responsive to the comparator.
14. The system of Claim 12, the comparator determining that the two pointers do not
15 reference the same data object if the SQUIDs are different.
15. The system of Claim 14, the comparator further determining that the two
pointers reference the same data object if the SQUIDs are identical and address
fields of the two pointers are identical.
16. The system of Claim 15, each pointer address field comprising:
20 a base address; and

FOIA b 7 - D

an offset,

the comparator further determining the two pointers do not reference identical locations within a referenced data object if the pointers' offsets are not identical.

- 5 17. The system of Claim 12, wherein a pointer is associated with a migration indicator field which indicates a number of migrations of the referenced data object by the time said pointer is created, the comparator determining that said two pointers do not reference the same data object if their associated migration indicators indicate identical numbers of migrations and their corresponding
10 addresses are different.
18. The system of Claim 17, wherein the migration indicator comprises one bit.
19. The system of Claim 1, wherein at least one pointer is a guarded pointer.
20. The system of Claim 1, wherein the SQUID is implemented by hardware.
21. The system of Claim 1, wherein the SQUID is implemented by software.
- 15 22. The system of Claim 1, wherein a pointer contains its associated SQUID.
23. The system of Claim 1, further comprising a SQUID cache for storing SQUIDS of recently-used pointers.
24. A data processing system for processing programs, the system comprising:
memory for storing data objects;
20 pointers to data objects stored in the memory;

migration indicators associated with pointers, a migration indicator indicating a number of migrations of a data object referenced by an associated pointer prior to said pointer being created; and

5 a comparator which determines that said two pointers do not reference the same data object if their associated migration indicators indicate identical numbers of migrations and their corresponding addresses are different.

25. The system of Claim 22, wherein the migration indicator comprises one bit.

26. The system of Claim 22, wherein the migration indicator comprises multiple bits.

10 27. The system of Claim 22, wherein the migration indicator is implemented by hardware.

28. The system of Claim 22, wherein the migration indicator is implemented by software.

29. A method for processing programs, the system comprising:
15 storing data objects in a memory, the data objects being referenced by pointers; and
generating a short-quasi-unique-identifier (SQUID) and assigning the SQUID to a data object stored in the memory segment, pointers to the data object being associated with the data object's assigned SQUID.

20 30. The method of Claim 27, further comprising:
allocating a segment of the memory to the data object.

31. The method of Claim 28, wherein if the data object is moved to a second allocated memory segment, a pointer to the second allocated memory segment is placed at the original memory segment.
32. The method of Claim 29, wherein the data object is moved due to resizing.
- 5 33. The method of Claim 29, wherein the data object is moved from a first memory to a second memory within a distributed system.
34. The method of Claim 29, wherein the data object is moved due to garbage collection.
35. The method of Claim 29, wherein the data object is moved due to data
10 compaction.
36. The method of Claim 27, wherein the distribution of SQUIDs over a range is uniform.
37. The method of Claim 34, wherein SQUIDs are generated by counting.
38. The method of Claim 34, wherein SQUIDs are generated randomly.
- 15 39. The method of Claim 34, wherein SQUIDs are generated by hashing.
40. The method of Claim 27, further comprising:
comparing SQUIDs of two different pointers.
41. The method of Claim 38, further comprising:
reordering instructions responsive to the comparison of SQUIDs.

F04T11B620660

42. The method of Claim 38, further comprising:
determining that the two pointers do not reference the same data object if the SQUIDs are different.
43. The method of Claim 40, further comprising:
5 determining that the two pointers reference the same data object if the SQUIDs are identical and address fields of the two pointers are identical.
44. The method of Claim 41, each pointer address field comprises a base address and an offset, the method further comprising:
10 determining the two pointers do not reference identical locations within a referenced data object if the pointers' offsets are not identical.
45. The method of Claim 38, wherein a pointer is associated with a migration indicator field which indicates a number of migrations of the referenced data object by the time said pointer is created, the method further comprising:
15 determining that said two pointers do not reference the same data object if their associated migration indicators indicate identical numbers of migrations and their corresponding addresses are different.
46. The method of Claim 43, wherein the migration indicator comprises one bit.
47. The method of Claim 27, wherein at least one pointer is a guarded pointer.
48. The method of Claim 27, wherein the SQUID is implemented by hardware.
- 20 49. The method of Claim 27, wherein the SQUID is implemented by software.
50. The method of Claim 27, wherein a pointer contains its associated SQUID.

51. The method of Claim 27, further comprising:
maintaining a SQUID cache for storing SQUIDS of recently-used
pointers.
52. A method for processing programs, the method comprising:
5 storing data objects in memory;
generating pointers to the stored data objects;
associating migration indicators with the pointers, a migration indicator
indicating a number of migrations of a data object referenced by an associated
pointer prior to said associated pointer being created; and
10 comparing two pointers and determining that said two pointers do not
reference the same data object if their associated migration indicators indicate
identical numbers of migrations and their corresponding addresses are different.
53. The method of Claim 48, wherein the migration indicator comprises one bit.
54. The method of Claim 48, wherein the migration indicator comprises multiple
15 bits.
55. The method of Claim 48, wherein migration indicators are implemented by
hardware.
56. The method of Claim 48, wherein the migration indicators are implemented by
software.
- 20 57. A data processing system for processing programs, the system comprising:
means for storing data objects in a memory, the data objects being
referenced by pointers; and

T04T"8620660

means for generating a short-quasi-unique-identifier (SQUID) and assigning the SQUID to a data object stored in the memory segment, pointers to the data object containing the data object's assigned SQUID.

58. The system of Claim 53, further comprising:

5 means for comparing SQUIDs of two different pointers.

59. A data processing system for processing programs, the method comprising:

means for storing data objects in memory;

means for generating pointers to the stored data objects;

10 means for associating migration indicators with the pointers, a migration indicator indicating a number of migrations of a data object referenced by an associated pointer prior to said associated pointer being created; and

15 means for comparing two pointers and determining that said two pointers do not reference the same data object if their associated migration indicators indicate identical numbers of migrations and their corresponding addresses are different.

60. A computer program product for processing programs, the computer program product comprising a computer usable medium having computer readable code thereon, including program code which:

20 stores data objects in a memory, the data objects being referenced by pointers; and

generates a short-quasi-unique-identifier (SQUID) and assigns the SQUID to a data object stored in the memory segment, pointers to the data object containing the data object's assigned SQUID.

61. The computer program product of Claim 56, further including program code which:

25

0990798-1401
FOHFF 86206660

compares SQUIDs of two different pointers.

62. A computer data signal embodied in a carrier wave for processing programs, comprising:
- 5 a program code segment for storing data objects in a memory, the data objects being referenced by pointers; and
- a program code segment for generating a short-quasi-unique-identifier (SQUID) and assigning the SQUID to a data object stored in the memory segment, pointers to the data object containing the data object's assigned SQUID.
- 10 63. The computer data signal of Claim 58, further comprising:
- a program code segment for comparing SQUIDs of two different pointers.
64. A computer program product for processing programs, the computer program product comprising a computer usable medium having computer readable code thereon, including program code which:
- 15 stores data objects in memory;
- generates pointers to the stored data objects;
- associates migration indicators with the pointers, a migration indicator indicating a number of migrations of a data object referenced by an associated
- 20 pointer prior to said associated pointer being created; and
- compares two pointers, determining that said two pointers do not reference the same data object if their associated migration indicators indicate identical numbers of migrations and their corresponding addresses are different.
65. A computer data signal embodied in a carrier wave for processing programs, comprising:
- 25

a program code segment for storing data objects in memory;

a program code segment for generating pointers to the stored data objects;

5 a program code segment for associating migration indicators with the pointers, a migration indicator indicating a number of migrations of a data object referenced by an associated pointer prior to said associated pointer being created; and

10 a program code segment for comparing two pointers, said segment determining that said two pointers do not reference the same data object if their associated migration indicators indicate identical numbers of migrations and their corresponding addresses are different.

0050.2025-001